Ottomater

USER MANUAL

Forward
Thinking
Designs

Version 1.0.0

Revised May 27, 2021

TABLE OF CONTENTS

OVERVIEW....................5

CONFIGURATION....................6

- Properties....................6
- Setup....................7

GETTING STARTED: PROPERTIES, SETTINGS, AND FUNDAMENTALS....................10

- Installation....................10
- Fundamentals....................10
  - Properties Panel....................11
  - Terminology....................11
- Using Ottomater....................12
  - About Cues....................12
  - About Commands....................14
  - Playback Controls....................17
  - Direct Cues Tab....................17
  - Configuration Tab....................17
- Conclusion....................18

GETTING STARTED: BUILD A SIMPLE SHOW....................19

- Design....................19
- Programming....................20
  - Start of Day....................20
  - End of Day....................22
  - Show Start....................22
  - Show Stop....................23
  - Light Cues....................24
- Running the Show....................26
- UCI Creation....................27
- Conclusion....................28

CONTROLS....................29

- Ottomater....................29
  - Playback Controls....................30
  - Cue List....................30
  - Commands....................31
  - Command Control List....................32
- Direct Cues....................33
- Configuration....................34

APPENDIX A: CONTROL PINS....................35

Direct Cue *35*
General *35*

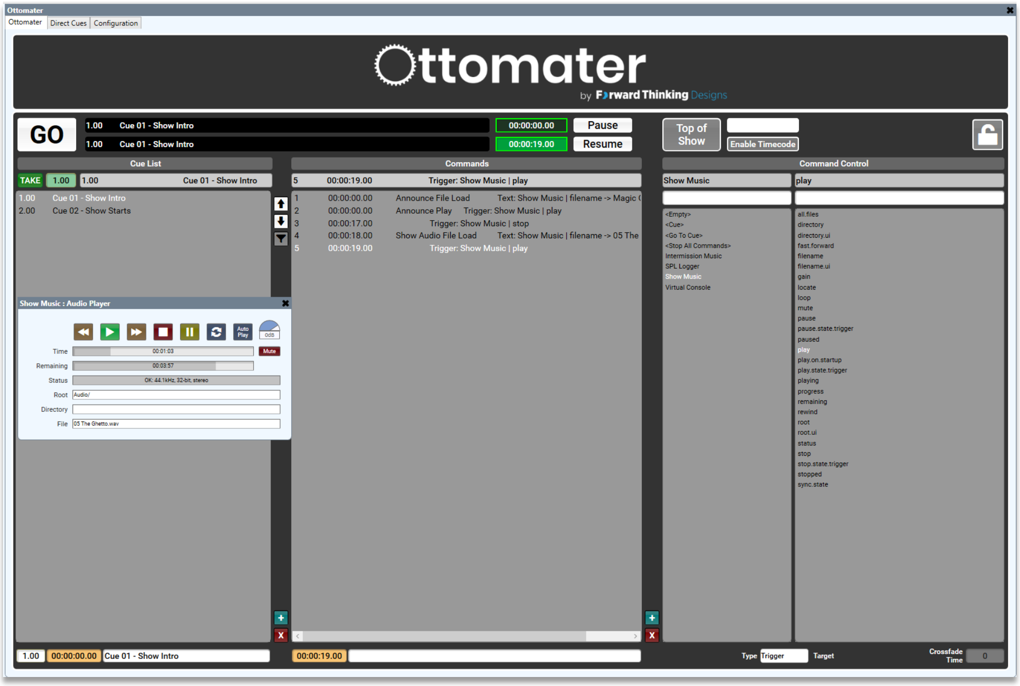
SUPPORT 36

OVERVIEW

Ottomater is a cue-based show control plugin for QSC Q-SYS Systems. Ottomater quickly and easily gives users control of any named component within their Q-SYS design allowing for ease of programming and flexibility in show design. This is ideal for both operator-controlled live performances and timecode based automated shows. With a simple, logical approach to building cues and firing commands, Ottomater is the smartest show control plugin out there for the QSC Q-SYS ecosystem.

Ottomater requires a license key to function but can be used in demo mode for 10 minutes to evaluate the plugin. The 10-minute period can be restarted by restarting the Q-SYS™ design. Use of the plugin in demo mode for commercial purposes is prohibited.

Ottomater downloads and license keys can be obtained from Forward Thinking Designs at ForwardThinkingDesigns.com.



1.00 Cue 01 - Show Intro 00:00:00.00
1.00 Cue 01 - Show Intro 00:00:19.00

Cue List: TAKE 1.00 | 1.00 Cue 01 - Show Intro

1.00	Cue 01 - Show Intro
2.00	Cue 02 - Show Starts

Commands: 5 00:00:19.00 Trigger: Show Music \| play

1	00:00:00.00	Announce File Load	Text: Show Music | filename -> Magic (
2	00:00:00.00	Announce Play	Trigger: Show Music | play
3	00:00:17.00		Trigger: Show Music | stop
4	00:00:18.00	Show Audio File Load	Text: Show Music | filename -> 05 The
5	00:00:19.00		Trigger: Show Music | play

Show Music : Audio Player
Time 00:01:03
Remaining 00:03:57
Status OK: 44.1kHz, 32-bit, stereo
Root Audio/
Directory
File 05 The Ghetto.wav

Command Control: Show Music | play

<Empty>
<Cue>
<Go To Cue>
<Stop All Commands>
Intermission Music
SPL Logger
Show Music
Virtual Console

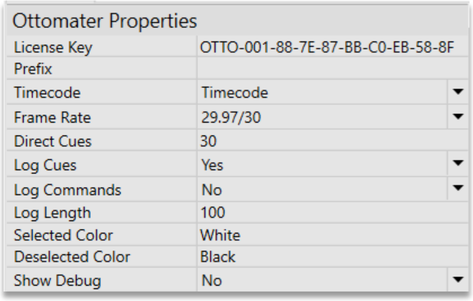
all.files
directory
directory.ui
fast.forward
filename
filename.ui
gain
locate
loop
mute
pause
pause.state.trigger
paused
play
play.on.startup
play.state.trigger
playing
progress
remaining
rewind
root
root.ui
status
stop
stop.state.trigger
stopped
sync.state

1.00 00:00:00.00 Cue 01 - Show Intro
00:00:19.00
Type Trigger Target
Crossfade Time 0

CONFIGURATION

Properties

Property	Function	Choices
License Key	Enter the license key here to activate the plugin.	
Prefix	Allows the user to filter which Named Components are available in Ottomater.	Manual Text Entry
Timecode	Sets the format of timecode received by Ottomater.	None Seconds Timecode Time of Day
Frame Rate	Sets the frame rate of the Timecode input. Available only if Timecode format is selected for the Timecode property.	23.98/24 25 29.97/30 Milliseconds
Direct Cues	Specifies the quantity of Direct Cue Triggers populated in Ottomater	0-90
Log Cues	Configures Ottomater to log when each cue is triggered to the internal log.	Yes No
Log Commands	Configures Ottomater to log when each command is executed to the internal log.	Yes No
Log Length	Configures the maximum length of the internal log.	10 – 500
Selected Color	Sets the color of text in the list box controls while selected.	Hex Color Value or HTML Color Name (i.e. #0CFF00, Red)
Deselected Color	Sets the color of text in the list box controls while not selected.	Hex Color Value or HTML Color Name (i.e. #0CFF00, Red)



Ottomater Properties

License Key	OTTO-001-88-7E-87-BB-C0-EB-58-8F
Prefix	
Timecode	Timecode
Frame Rate	29.97/30
Direct Cues	30
Log Cues	Yes
Log Commands	No
Log Length	100
Selected Color	White
Deselected Color	Black
Show Debug	No

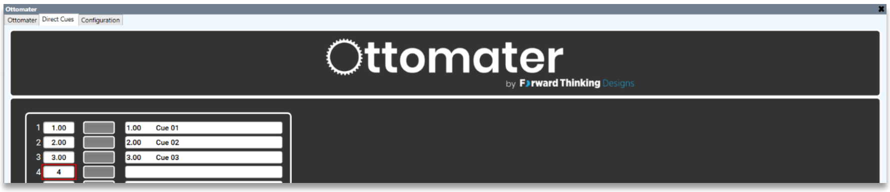
Setup

To configure Ottomater, follow these steps:

1. Enter your License Key in the Properties window. Without a license the plugin will only function for 10 minutes before your design must be restarted.
2. To limit the Named Components controlled by Ottomater, designate a Prefix in the **Prefix** property.

   This setting allows you to limit which Named Components are displayed in and controlled by Ottomater. This is useful if your Q-SYS design has many Named Components or you have multiple instances of Ottomater running in a single Q-SYS design. For example, if your design has Stage A and Stage B, if you enter "Stage A-" into prefix that instance will only populate named controls that start with "Stage A-", preventing accidental control of the wrong components.

3. If Ottomater is using external timecode or clock, select the appropriate **Timecode** format in the plugin Properties.
   a. **None** for a show doesn't use Timecode to control Ottomater.
   b. **Seconds** accepts timecode as a floating-point number of seconds. This is the format output by the SMPTE Reader component Time control pin and accepted by the Timeline Player component External Time control pin.
   c. **Timecode** is the correct format for a design that accepts timecode as a string and requires frame-level timing. The string will be accepted in the format HH:MM:SS.FF (Hours:Minutes:Seconds.Frames).
      i. Selecting Timecode makes the **Frame Rate** property available. Choices are 23.98/24, 25, 29.97/30, and milliseconds.
   d. **Time of Day** is the correct format for a design that accepts Timecode as a String and requires seconds-level timing. The string will be accepted in the format HH:MM:SS.
4. If your design requires Direct Cues, set the **Direct Cues** property to the desired number. Ottomater will create this number of buttons on the Direct Cues page.



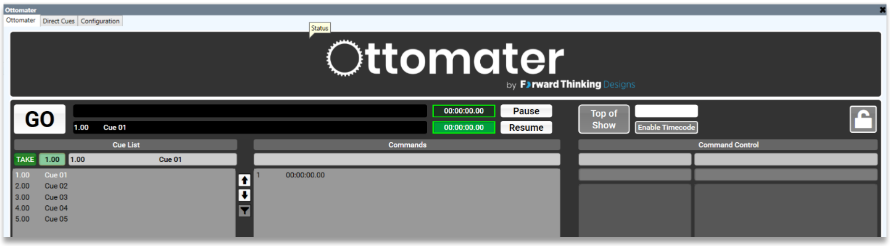
1 1.00 — 1.00 Cue 01
2 2.00 — 2.00 Cue 02
3 3.00 — 3.00 Cue 03
4 4

5. Configure Ottomater's internal log, set the **Log Cues** property to log the triggering of each cue and the source. Set the **Log Commands** property to log the execution of each command and the cue it originated from. Logging commands is very useful during programing and diagnosis but

can quickly fill up the log, it's recommended to turn off command logging during normal cue playback.

   a. Set the value of the **Log Length** property to the desired maximum number of log entries, from 10 up to 500.

6. Enter the desired colors for **Selected Text** and **Deselected Text** in the Properties window. In the example below, the Selected Text (the next active cue, in this case) is white and the Deselected Text (all other text) is black. This is useful if Ottomater will be placed on a UCI with a color scheme that differs from the default.



1.00 Cue 01

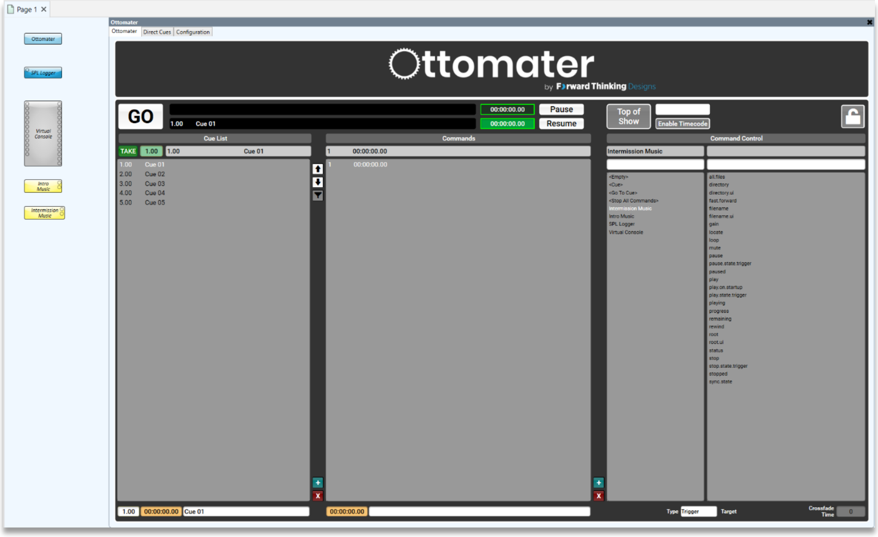
00:00:00.00
00:00:00.00

1.00 | 1.00 Cue 01

1.00 Cue 01
2.00 Cue 02
3.00 Cue 03
4.00 Cue 04
5.00 Cue 05

1 00:00:00.00

7. In Q-SYS Designer, give a unique and recognizable name to every component that will be controlled by Ottomater. These component names will be populated in the Command Control panel of Ottomater. As in the example below, all design components with a unique name are populated in the Command Control section, and all controls for those components are available to be controlled.



1.00 Cue 01

00:00:00.00
00:00:00.00

1.00 | 1.00 Cue 01

1.00 Cue 01
2.00 Cue 02
3.00 Cue 03
4.00 Cue 04
5.00 Cue 05

1 00:00:00.00
1 00:00:00.00

Intermission Music

<Empty>
<Cue>
<Go To Cue>
<Stop All Commands>
Intermission Music
Intro Music
SPL Logger
Virtual Console

all.files
directory
directory.ui
fast.forward
filename
filename.ui
gain
locate
loop
mute
pause
pause.state.trigger
paused
play
play.on.startup
play.state.trigger
playing
progress
remaining
rewind
root
root.ui
status
stop
stop.state.trigger
stopped
sync.state

1.00 | 00:00:00.00 | Cue 01

00:00:00.00

Type Trigger

Crossfade Time 0

8. The offline plugin settings are now finished, and the plugin should be run on an active Q-SYS™ Core using *Save to Core & Run*.

9. Congratulations, Ottomater is now ready to be programmed!

GETTING STARTED: PROPERTIES, SETTINGS, AND FUNDAMENTALS

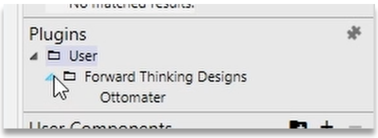
Ottomater allows you to cue any control within your design, including third-party plugins, all without the need to write a single line of code. You can fire any number of cues in sequence using the Go button, or sync a Cue using external timecode, time of day, or even dedicated, out-of-sequence triggers.

It is highly recommended to review the Ottomater Setup steps before beginning this Getting Started chapter. This chapter will cover Ottomater's properties, settings, and basic use.

Installation

Download the Ottomater plugin from ForwardThinkingDesigns.com and double-click the Ottomater.qplugx file to install. Click “Yes” in the dialog box to confirm the installation.

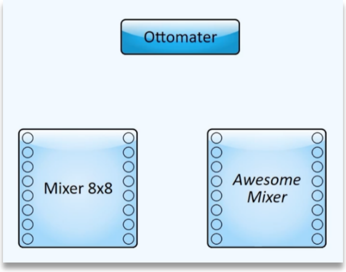
Ottomater will now appear in Q-SYS Designer in your designer plugins directory under User > Forward Thinking Designs. Drag Ottomater into your design.



If you purchased a license key, copy and paste that key into the license key property of the plugin. Without a license key, full operation is limited to a 10-minute demo mode.

Fundamentals

Ottomater works by controlling Named Components in your Q-SYS design. In this example, there are two mixers, one of which is named “Awesome Mixer.” Awesome Mixer will be available to Ottomater for control because it has been named. However, the unnamed mixer will not be available. Make sure anything that should be controlled in Ottomater is given a unique name in your design.



Properties Panel

The Properties Panel is explained in greater detail in the Configuration > Setup chapter of this manual.

- Prefix determines which named components show up in Ottomater. For example, if Prefix is set to "Otto", only named components beginning with the string 'Otto' will appear in Ottomater, even if there are hundreds of named components in the design.
- Timecode determines the format of the time source triggering cues.
- Direct Cues determines the number of Direct Cue Triggers to create.
- Log Cues determines whether or not to write to the log each time a Cue is triggered.
- Log Commands determines whether or not to write to the log each time a Command is triggered.
- Log Length determines how many log entries to keep.
- Selected Color and Deselected Color determine the look of Ottomater's list boxes.

Terminology

Ottomater is organized hierarchically.

- Show – The ordered collection of Cues in Ottomater. A Show can also be thought of like the script for Ottomater.
  - Cue – A container for ordered sequences of Commands. Cues can be fired manually or by Timecode. A Cue often corresponds to a "beat" in the performance, or is otherwise a collection of Commands that are executed together. When a Cue is triggered, each Command within that Cue will play in sequence.
    - Command – An action taken on one Control in one Component. Examples of Commands include setting a button true, changing a fader to 0dB, or even firing off another Cue.

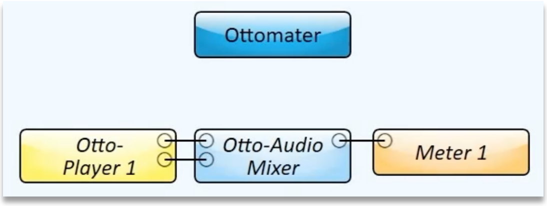
  - Component – A Named Component in Q-SYS outside of Ottomater, including 3rd party plugins, such as an Audio Player, Matrix Mixer, or Gain component.
    - Control – A single Q-SYS Control within a Component. This may be a button, knob, textbox or others.

Commented [AB1]: Get rid of this underline

- Trigger – Anything that starts a Cue. This can be the Go button, a Direct Cue button, any of the available time options or another cue.
- Cue Number – A numerically sorted address for the Cue expressed to two decimal places. The Cue List is automatically sorted by Cue Number.
- Cue Timecode – The time at which a Cue should be automatically triggered by Timecode.
- Cue Name – The descriptive name of the Cue.
- Direct Cue – A method for at-will firing of a given Cue, independent of the sequence of the Cues in the Cue List.

Using Ottomater

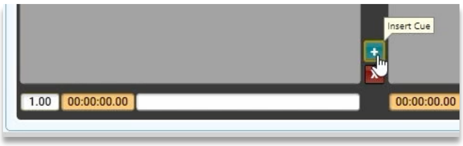
For this example, assume that the Q-SYS design already has some named components.



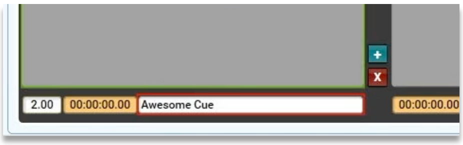
Run the design, and open Ottomater.

About Cues

First, add a Cue. Click on the Insert Cue button (the plus button to the lower right of the Cue List) to create the first Cue.

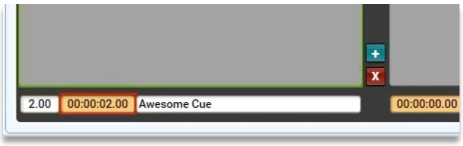


The Cue Name and Cue Number can be changed by clicking and typing in the Cue Name and Edit Cue Number fields at the bottom of the Cue List section. In this example, the Cue is changed to Cue 2, and renamed to Awesome Cue.



+
X
2.00
00:00:00.00
Awesome Cue
00:00:00.00

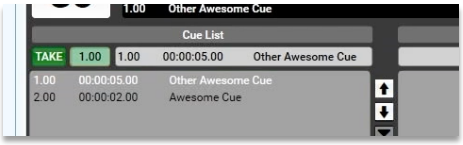
If the Cue is to be triggered by Timecode, change the Cue Timecode to anything other than zero (using the Edit Cue Timecode field at the bottom left of the Cue List section. Ottomater will then treat this as a Timecode-enabled Cue. In this example, this Cue's Timecode is set to two seconds.



2.00 | 00:00:02.00 | Awesome Cue | 00:00:00.00

The Timecode will now appear in the Cue List box next to the Cue Name and Cue Number. The trailing decimal time number format here is determined by the Timecode Property. For Timecode input in Seconds, this value is hundredths of a second. For Timecode input in Timecode, this value is in frames. For Clock, the trailing decimal is not used, and the clock only tracks whole seconds.

Next, add another Cue. In this example, the second Cue is named Another Awesome Cue, and has a Cue Timecode of five seconds, while the Cue Number is changed to Cue 1. In this scenario, Awesome Cue (two seconds) will be triggered before Another Awesome Cue (five seconds) via Time Code. However, Another Awesome Cue (Cue 1) appears before Awesome Cue (Cue 2) in the Cue List. Thus, manually triggering the Cues will have one order, whereas Timecode triggers will have a different order.

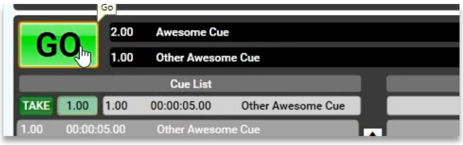

1.00 Other Awesome Cue

Cue List

TAKE 1.00 | 1.00 | 00:00:05.00 | Other Awesome Cue

1.00	00:00:05.00	Other Awesome Cue
2.00	00:00:02.00	Awesome Cue

Click the Go button to play the currently selected Cue. Once a Cue is triggered, the next Cue will automatically be selected. There are Recall Cue and Take buttons which can be wired to an external control script for direct access.



2.00 Awesome Cue

1.00 Other Awesome Cue

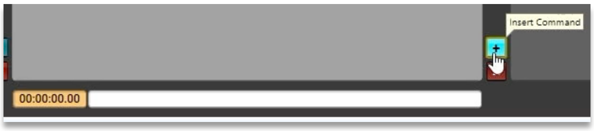
Cue List

TAKE 1.00 | 1.00 | 00:00:05.00 | Other Awesome Cue

1.00 00:00:05.00 Other Awesome Cue

About Commands

These Cues are currently empty, so nothing is being controlled. Create a Command now by clicking the Insert Command button at the bottom right of the Commands List. The Command will be added to whichever Cue is currently selected.



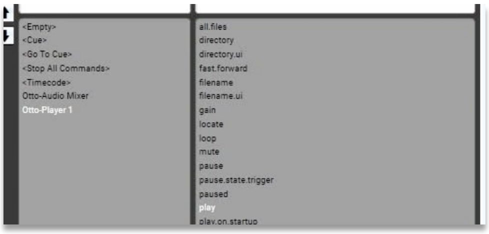
00:00:00.00

The Command Name can be set by clicking and typing in the Command Name field, located at the bottom of the Commands List. This name should describe exactly what task the Command performs.

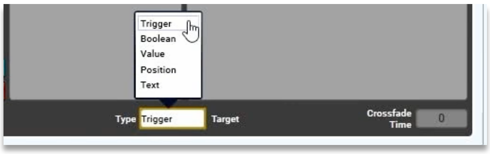
Command Time is expressed as a delay from the beginning of the Cue. For example, if a Command has a Command Time of 00:00:07.00, that Command will execute 7 seconds after the Cue is triggered. By default, Commands are created with no delay (00:00:00.00), these commands will be executed as soon as the Cue is triggered. The command delay can be changed by clicking and typing in the Edit Command Time field at the bottom left of the Commands List. When editing Command Times, the trailing decimal time number is always expressed in hundredths of a second.

While a Command is selected in the Commands List, the Command Control List will display a list of internal Ottomater functions and Named Components that can be controlled by the selected Command. The left column is populated with the Named Components in the Q-SYS design. The right column is populated with the Controls of the selected Component.

In this example, select “Otto-Player 1” and then select “play” from the list of Controls in “Otto-Player 1”.



At the bottom of the Command Control List are Type, Target, and Crossfade Time.



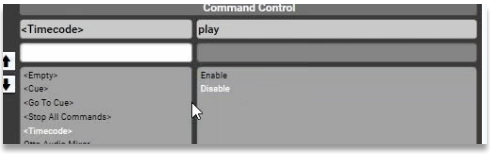
Type Trigger
Target
Crossfade Time 0

- Type – The type of the control: Trigger, Boolean, Value, Position or Text. While it is possible in Q-SYS to change Controls using other Control Types, it is always recommended to use the native Control Type to prevent unexpected behavior. If you are unfamiliar with Control Types explore QSC’s online Control 101 training.
- Target – This is the Target value to change the Control to. For a Control Type of Boolean, the options are On and Off, for a Control Type of Value and Position, the Target is a number and for a Control Type of Text the Target is a string.
- Crossfade Time – Value and Position Control Types can be cross faded from their current value, to the Target value. Set the length of the fade in seconds, a Crossfade Time of 0 is an instant change.

In the example the Play control is a Trigger, a trigger has no Target value and cannot be crossfaded.

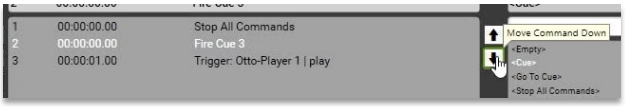
Ottomater has certain internal functions which can be used in Commands just like Components.

- <Timecode> can be set to Enable or Disable. Disabling Timecode prevents cues from being triggered by Timecode.



- <Stop All Commands> will stop all commands that Ottomater is running, except for the ones in the same Cue. For example: If Cue 1 is running and <Stop All Commands> is used in Cue 2, that will stop any commands that are still running in Cue 1, and then proceed to run any additional Commands in Cue 2.

Commented [AB2]: Update this image to reflect the new text order



1 00:00:00.00 Stop All Commands
2 00:00:00.00 Fire Cue 3
3 00:00:01.00 Trigger: Otto-Player 1 | play

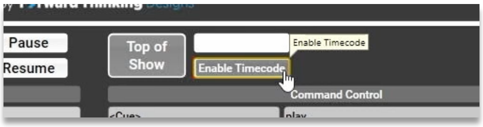
- <Stop Cue> will stop a specific numbered Cue.
- <Go To Cue> sets the currently selected Cue to a designated Cue. For example, if Cue 1 contains <Go To Cue> (Cue 3), triggering Cue 1 will cause Ottomater to select Cue 3.
- <Cue> will trigger the selected Cue, executing the Commands in that cue.

Commands will always execute in the order that they appear in the Command List. To change the order of Commands, use the Move Command Up and Move Command Down buttons to reorder the selected Command. Only Commands that happen at the same Time as one another can be ordered in this way. For example: A Command with a Time of 00:00:00.00 can be moved up or down past another Command with Time 00:00:00.00. It cannot be moved past a Command with Time 00:00:01.00.

Commands can be deleted by selecting the Command and clicking Delete Command.

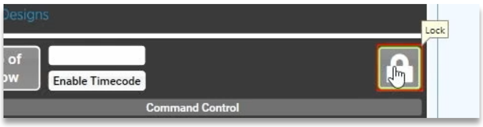
Playback Controls

Cues can be triggered by the GO button, Direct Cue Button or by Timecode. When the operator presses the GO button, the currently selected Cue will play, and the next Cue (if any) will be selected. If Timecode is enabled, a Cue is triggered automatically when the Timecode clock reaches that Cue's Cue Time. The Enable Timecode button at the top of the screen enables and disables the triggering of Cues by Timecode.



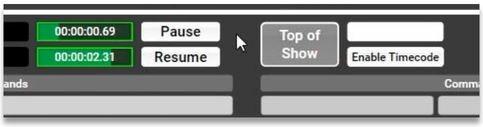
To change the Cue selection to the first Cue, click Top of Show.

Once the show is configured, the Lock button will disable the editing controls. Cues can still be triggered while Ottomater is locked, but Commands cannot be accidentally changed while the show is in progress.



The Pause and Resume buttons Pause and Resume playback of the show. Note that using the Go or Take buttons will also resume playback.

Time Elapsed and Time Remaining are shown next to Pause and Resume. These show the time elapsed and time remaining of the last Cue triggered.



Direct Cues Tab

Direct Cues are a method for firing a Cue without first selecting it.

Direct Cues are associated with Cues by Cue Number, entered in the left column. This will populate the name of the Cue in the right column, and connect the Direct Cue Take button in the center column with the Cue itself.

Configuration Tab

The Show File is displayed on the Configuration page, where it keeps a record of the show in a text-based format. This can be copied and pasted to save the file outside of Ottomater or into another instance of Ottomater.

The Log shows Cues and Commands triggered by Ottomater, subject to the Log Length specified in Properties. The detail also includes time stamps and what triggered the event. In this example, the Log is comprised of the example Cues presented in this guide.

Cue List Limit sets the maximum number of Previous Cues and Next Cues to display at once. This feature was intended for instances where the Cue List is placed on a UCI and fewer than the entire list should be displayed at once.

Conclusion

We hope that this Getting Started guide has been helpful and instructive. In the next section, we will walk through the creation of an entire show.

GETTING STARTED: BUILD A SIMPLE SHOW

In this walkthrough, we will create a simple show in Ottomater, starting with a blank show file. This goes into substantial detail, so this walkthrough will use the following format to concisely describe Commands in Ottomater:

[Time (if any) Type: Component | Control to Target(if any) in Crossfade Time(if any)]

So for example, a Trigger command with no Time delay would look like this:

[Trigger: BGM Player | play]

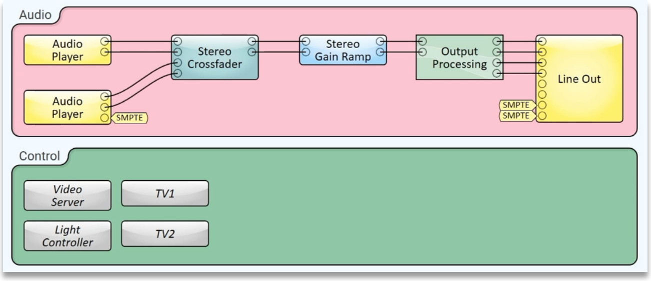
and delayed gain adjustment would look like this:

[00:00:02.00 Value: System Mute | gain to 0.5 in 3]

Design

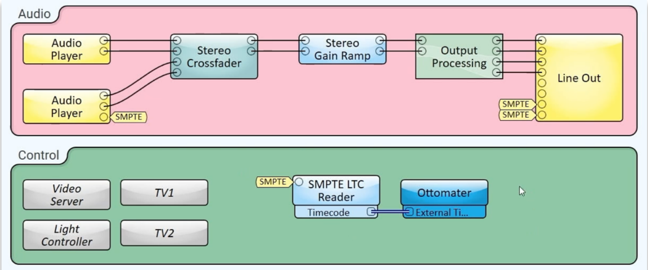
This example begins with a very basic design. Audio Components are a BGM player, a show audio player, a crossfader to fade between the two, a gain ramp (to mute the system), some output processing, and then a line out.

Control Components are a video server, a light controller, and two TVs to turn on and off.

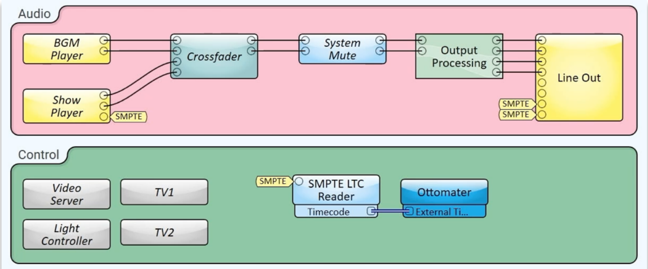


First, add Ottomater to the design. This example will also use SMPTE to sync cues to the show audio player, so add a SMPTE LTC Reader also. Enable Timecode Output for the SMPTE LTC Reader. For Ottomater, set the Timecode Property to Timecode and the Frame Rate Property to 30 frames per second.

Enable the External Time Code pin for Ottomater. Wire or flag the SMPTE signal from the Audio Player to the SMPTE LTC Reader, and wire the Reader Timecode pin to Ottomater's Timecode pin. Now Timecode will come into Ottomater.



Next, name the components to be controlled in Ottomater. The Audio Players should be descriptively named "BGM Player" and "Show Player". The Stereo Crossfader is called "Crossfader" and the Stereo Gain Ramp, "System Mute".



Now, make sure that Ottomater has a license key, and start the design. Once the design is running, Ottomater is ready for programming.

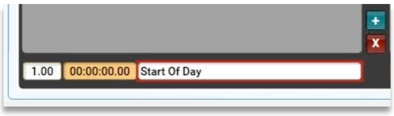
Programming

There are a few requirements or design goals for this example.

- Start of Day, to turn on the TVs, trigger the video server, trigger the light controller, and start the BGM playback.
- End of Day, to stop all audio playback and turn off the TVs.
- Show Start, to fade the crossfader over to the show player, and trigger the video server and the light controller.
- Show Stop, to return to BGM playing.

Start of Day

Begin by inserting a Cue and naming it “Start of Day”.



1.00 | 00:00:00.00 | Start Of Day

The first thing done in the day should be to turn on the TVs. The TV Components here are command buttons, and the Control is mapped to turn on the TV. Create two Commands and use Triggers to turn on the TVs.

[Trigger: TV1 | command.1.trigger]

[Trigger: TV2 | command.1.trigger]

Next, create a Command to unmute the audio.

[Boolean: System Mute | mute to OFF]

Create Commands to set the crossfader to BGM, load the correct file via the filename Control, and begin playing the BGM

[Boolean: Crossfader | crossfade.to.A to ON]

[Text: BGM Player | filename to “BGM Track.mp3”]

[Trigger: BGM Player | play]

In the event that the show audio is playing for some reason, add a Command to stop it.

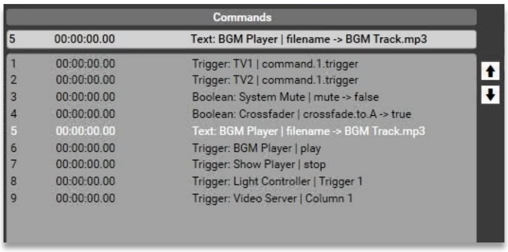
[Trigger: Show Player | stop]

Lighting and media playback are handled similarly. Add a Command to start the Light Controller and Video Server.

[Trigger: Light Controller | Trigger1]

[Trigger: Video Server | Column1]

The Command List for the Start of Day should now look like this:

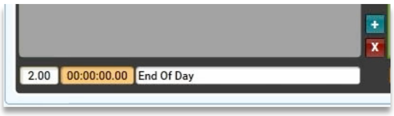


5	00:00:00.00	Text: BGM Player | filename -> BGM Track.mp3
1	00:00:00.00	Trigger: TV1 | command.1.trigger
2	00:00:00.00	Trigger: TV2 | command.1.trigger
3	00:00:00.00	Boolean: System Mute | mute -> false
4	00:00:00.00	Boolean: Crossfader | crossfade.to.A -> true
5	00:00:00.00	Text: BGM Player | filename -> BGM Track.mp3
6	00:00:00.00	Trigger: BGM Player | play
7	00:00:00.00	Trigger: Show Player | stop
8	00:00:00.00	Trigger: Light Controller | Trigger 1
9	00:00:00.00	Trigger: Video Server | Column 1

The Start of Day Cue is complete.

End of Day

The End of Day Cue will be similar to the Start of Day Cue, but in reverse. Begin by adding a Cue and naming it “End of Day”.



2.00 | 00:00:00.00 | End Of Day

Add a Command that uses the System Mute to fade the audio out. Note that the mute Control used here will take about two seconds to complete.

[Boolean: System Mute | mute to ON]

Now, add two more Commands for Show Player and BGM Player, triggering the stop Control for each of these after a two second delay. A two second delay is appropriate because the previous action takes about that amount of time to complete.

[00:00:02.00 Trigger: BGM Player | stop]

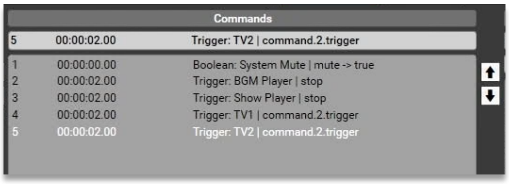
[[00:00:02.00 Trigger: Show Player | stop]

Finally, turn off the TVs. Create another two Commands, one for TV1 and one for TV2. Trigger the command.2.trigger Control on each Component.

[00:00:02.00 Trigger: TV1 | command.2.trigger]

[00:00:02.00 Trigger: TV2 | command.2.trigger]

The Command List for the End of Day should now look like this:



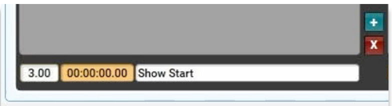
Commands

5	00:00:02.00	Trigger: TV2 | command.2.trigger
1	00:00:00.00	Boolean: System Mute | mute -> true
2	00:00:02.00	Trigger: BGM Player | stop
3	00:00:02.00	Trigger: Show Player | stop
4	00:00:02.00	Trigger: TV1 | command.2.trigger
5	00:00:02.00	Trigger: TV2 | command.2.trigger

With this, all the systems are stopped, and the End of Day Cue is now complete.

Show Start

The Show Start is next. Create another Cue and name it “Show Start”.



3.00 00:00:00.00 Show Start

The first thing to do at the start of the show will be to stop anything else that is running in the background. Add a Command to the new Cue that will stop all running Commands.

[Stop All Commands]

Next, in order to begin playing show audio and crossfade it in, add three more Commands. The first Command should load the correct show file. The seconds Command will cause the Show Player Component to begin playing. The third Command will instruct the Crossfader to fade to the Show Player.

[Text: Show Player | filename to “Show LTC 01-00-00_00.wav”]

[Trigger: Show Player | play]

[Boolean: Crossfader | crossfade.to.B to ON]

Add another two commands, starting the Video Server and Light Controller. Note that our video server will need a couple of frames to grab the empty signal, so we’re going to delay the Video Server by half a second to avoid a black screen.

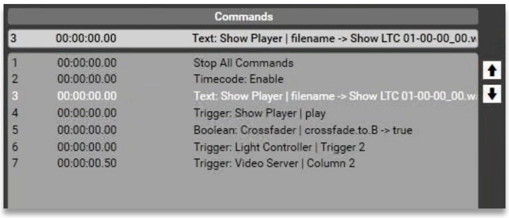
[Trigger: Light Controller | Trigger 2]

[00:00:00.50 Trigger: Video Server | Column 2]

Finally, enable Timecode and move it to right after Stop All Commands so it will run at the beginning of the Cue. This will allow Timecode to trigger Cues in the Show.

[Timecode: Enable]

The Command List for Show Start should now look like this:



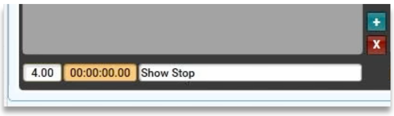
Commands

3	00:00:00.00	Text: Show Player | filename -> Show LTC 01-00-00_00.w
1	00:00:00.00	Stop All Commands
2	00:00:00.00	Timecode: Enable
3	00:00:00.00	Text: Show Player | filename -> Show LTC 01-00-00_00.w
4	00:00:00.00	Trigger: Show Player | play
5	00:00:00.00	Boolean: Crossfader | crossfade.to.B -> true
6	00:00:00.00	Trigger: Light Controller | Trigger 2
7	00:00:00.50	Trigger: Video Server | Column 2

The Show Start Cue is finished and it’s time to move on to the next Cue.

Show Stop

The show needs a Show Stop Cue. The Show Stop Cue will begin by stopping any other currently running commands. Create a Cue and name it “Show Stop”.



4.00 00:00:00.00 Show Stop

[Stop All Commands]

Return the Crossfader to A, and load the correct file into the BGM Player. Then, begin playing that file with the BGM Player. The Light Controller needs to return to Trigger 1, and the Video Server should be returned to Column 1.

[Boolean: Crossfader | crossfade.to.A to ON]

[Text: BGM Player | filename to “BGM Track.mp3”]

[Trigger: BGM Player | play]

[Trigger: Light Controller | Trigger 1]

[Trigger: Video Server | Column 1]

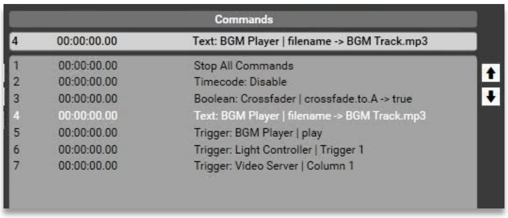
Add a Command to stop the playback of the show file.

[00:00:02.00 Trigger: Show Player | stop]

Finally, add a Command to disable Timecode, which will prevent any additional Cues from running when the show is stopped. This should actually be placed right after Stop All Commands, so move it to that position using the Move Command Up button.

[Timecode: Disable]

The Command List for Show Stop should look like this:



Commands

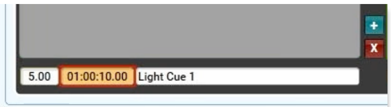
4	00:00:00.00	Text: BGM Player | filename -> BGM Track.mp3
1	00:00:00.00	Stop All Commands
2	00:00:00.00	Timecode: Disable
3	00:00:00.00	Boolean: Crossfader | crossfade.to.A -> true
4	00:00:00.00	Text: BGM Player | filename -> BGM Track.mp3
5	00:00:00.00	Trigger: BGM Player | play
6	00:00:00.00	Trigger: Light Controller | Trigger 1
7	00:00:00.00	Trigger: Video Server | Column 1

The Show Stop Cue is finished, and it’s time to create the show Cues.

Light Cues

Now that the start and end of the show and the day are programmed, it’s time to add some lighting cues to the show. These Cues will all run after the Show Start, and Timecode will be running, so these can all be triggered by Timecode.

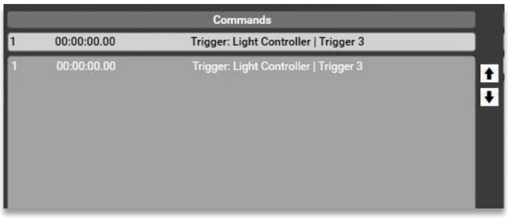
Add another Cue to the show, and name it Light Cue 1. Give it a Timecode of 01:00:10.00 (starting time of one hour and ten seconds).



5.00 | 01:00:10.00 | Light Cue 1

The first Command to add to Light Cue 1 is simply triggering a Light Controller.

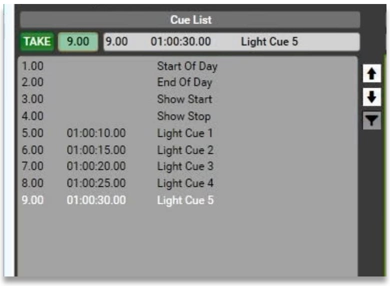
[Trigger: Light Controller | Trigger 3]



Commands

1	00:00:00.00	Trigger: Light Controller | Trigger 3
1	00:00:00.00	Trigger: Light Controller | Trigger 3

Add four more Light Cues to the show, named Light Cue 2 through Light Cue 5. Light Cue 2 will start five seconds after Light Cue 1, and so on.



Cue List

TAKE | 9.00 | 9.00 | 01:00:30.00 | Light Cue 5

1.00		Start Of Day
2.00		End Of Day
3.00		Show Start
4.00		Show Stop
5.00	01:00:10.00	Light Cue 1
6.00	01:00:15.00	Light Cue 2
7.00	01:00:20.00	Light Cue 3
8.00	01:00:25.00	Light Cue 4
9.00	01:00:30.00	Light Cue 5

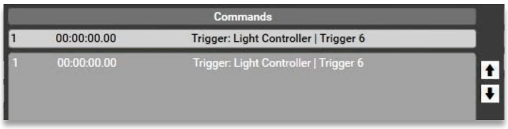
Also, add appropriate Light Controller commands to the different Light Cues (only one is pictured):

[Trigger: Light Controller| Trigger 4]

[Trigger: Light Controller| Trigger 5]

[Trigger: Light Controller| Trigger 6]

[Trigger: Light Controller| Trigger 7]

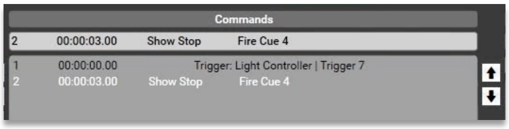


Commands

1	00:00:00.00	Trigger: Light Controller | Trigger 6
1	00:00:00.00	Trigger: Light Controller | Trigger 6

Return to Light Cue 5 and add a Command to fire Cue 4. This will stop the show after the last Light Cue. Delay it by three seconds and give it the name, “Show Stop”.

[00:00:03.00 Fire Cue 4]



Commands

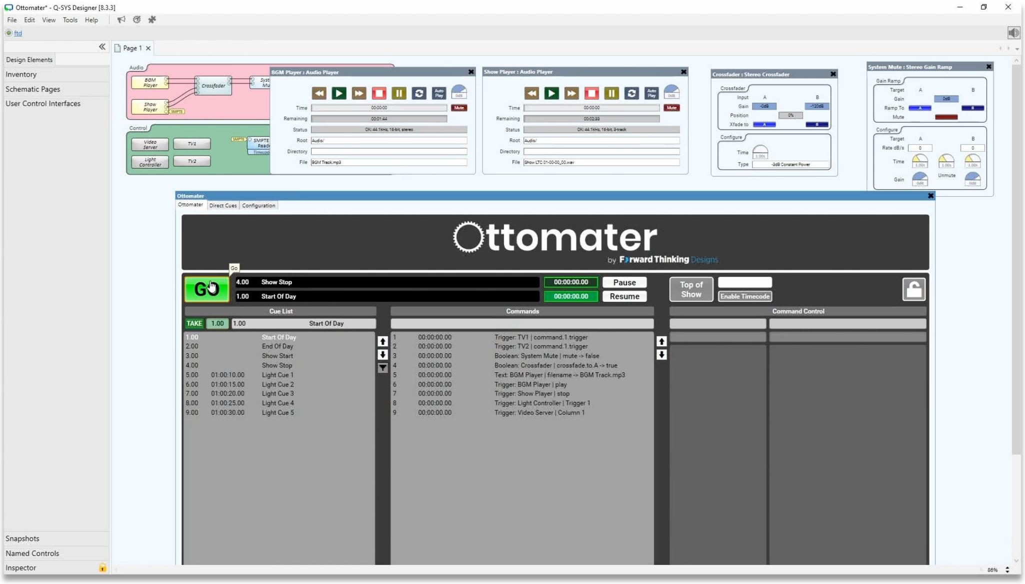
2	00:00:03.00	Show Stop	Fire Cue 4
1	00:00:00.00	Trigger: Light Controller | Trigger 7	
2	00:00:03.00	Show Stop	Fire Cue 4

Now, the final show Cue will trigger the Show Stop Cue.

This concludes the programming portion of the walkthrough. Next it's time to run the show.

Running the Show

Now that the show is programmed, run the Start of Day Cue. Notice that the BGM Player has started playing, the Crossfader has faded to input A, and the System Mute is unmuted. However, Timecode has not begun to run since nothing has enabled it.



4.00	Show Stop
1.00	Start Of Day

00:00:00.00
00:00:00.00

Cue List

1.00		Start Of Day
2.00		End Of Day
3.00		Show Start
4.00		Show Stop
5.00	01:00:10.00	Light Cue 1
6.00	01:00:15.00	Light Cue 2
7.00	01:00:20.00	Light Cue 3
8.00	01:00:25.00	Light Cue 4
9.00	01:00:30.00	Light Cue 5

Commands

1	00:00:00.00	Trigger: TV1 | command.1.trigger
2	00:00:00.00	Trigger: TV2 | command.1.trigger
3	00:00:00.00	Boolean: System Mute | mute -> false
4	00:00:00.00	Boolean: Crossfader | crossfade.to.A -> true
5	00:00:00.00	Text: BGM Player | filename -> BGM Track.mp3
6	00:00:00.00	Trigger: BGM Player | play
7	00:00:00.00	Trigger: Show Player | stop
8	00:00:00.00	Trigger: Light Controller | Trigger 1
9	00:00:00.00	Trigger: Video Server | Column 1

Now, run the Show Start Cue by selecting the Show Start Cue and clicking Go. Notice that the Show Player has started, the Crossfader has faded to input B, and Timecode is running.

Additionally, Ottomater is taking the Light Cues in sync with the Timecode clock, because Timecode is enabled. Once Light Cue 5 takes, the show will stop a few seconds afterwards.

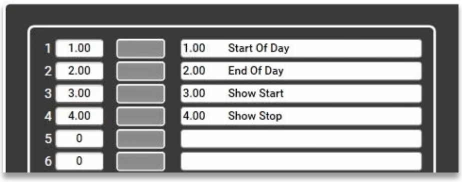
Once the show has stopped, the Crossfader will fade back to the BGM, the Show Player will stop running, and the lights and video will return to normal.

At this point, all that remains is to build a UCI.

UCI Creation

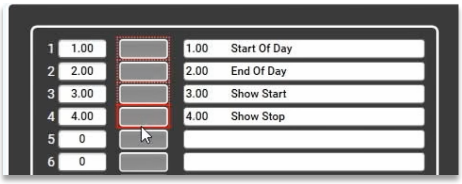
The show is programmed and running as desired, but currently has to be run through Ottomater. This isn't always ideal, so it's time to put the controls on a UCI. Navigate to the Direct Cues tab.

The major commands in use during the show are 1, 2, 3, and 4. Enter those Cue numbers into the first four Direct Cue Numbers.



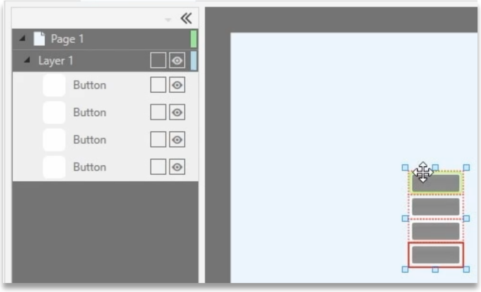
1	1.00	1.00	Start Of Day
2	2.00	2.00	End Of Day
3	3.00	3.00	Show Start
4	4.00	4.00	Show Stop
5	0		
6	0		

Stop the design, and copy the buttons associated with those Cues.

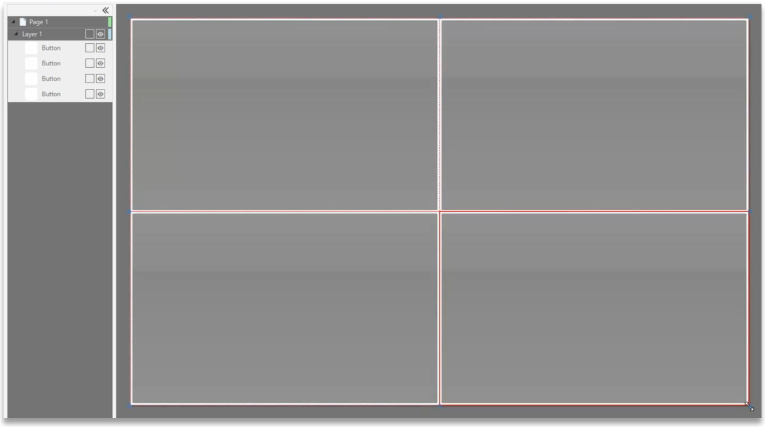


1	1.00	1.00	Start Of Day
2	2.00	2.00	End Of Day
3	3.00	3.00	Show Start
4	4.00	4.00	Show Stop
5	0		
6	0		

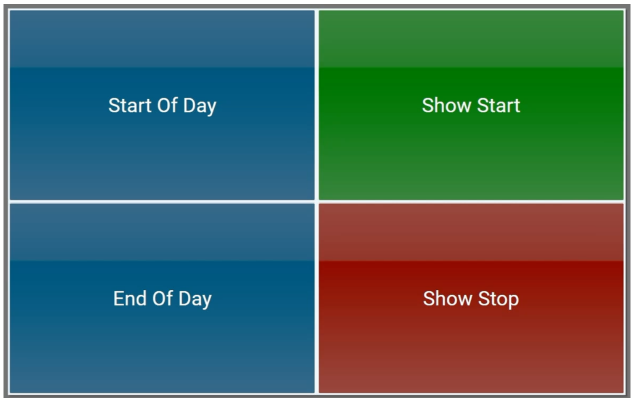
Create a new UCI, call it "Show Control", and paste the Direct Cue Buttons into the Show Control UCI.



Rearrange and resize the buttons to fit the UCI.



Name each of the buttons according to their respective Cues – “Start of Day”, “End of Day”, “Show Start”, and “Show Stop”. Next, resize the text to be more readable, and set the colors for these buttons to suit their function – green for Show Start, red for Show Stop, and blue for Start of Day and End of Day.



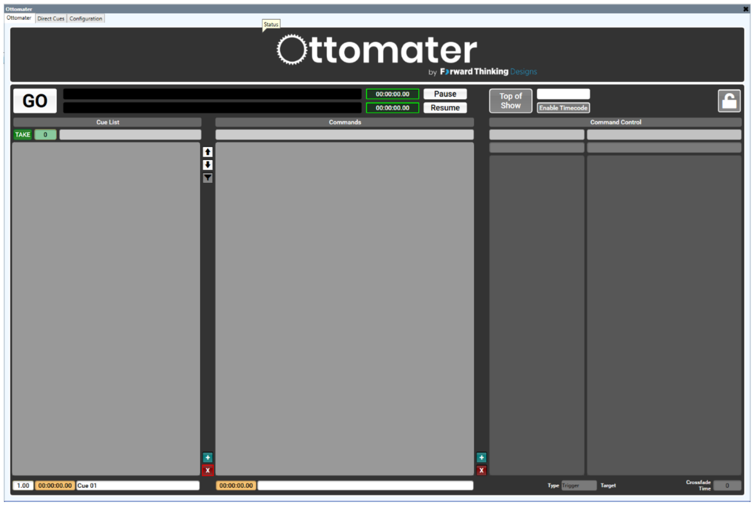
Now run the design again. The Start of Day Button runs the Start of Day, Show Start causes the show to run, and Show Stop stops the show at any point. End of Day will mute the system and stop all the players. The UCI is now complete.

Conclusion

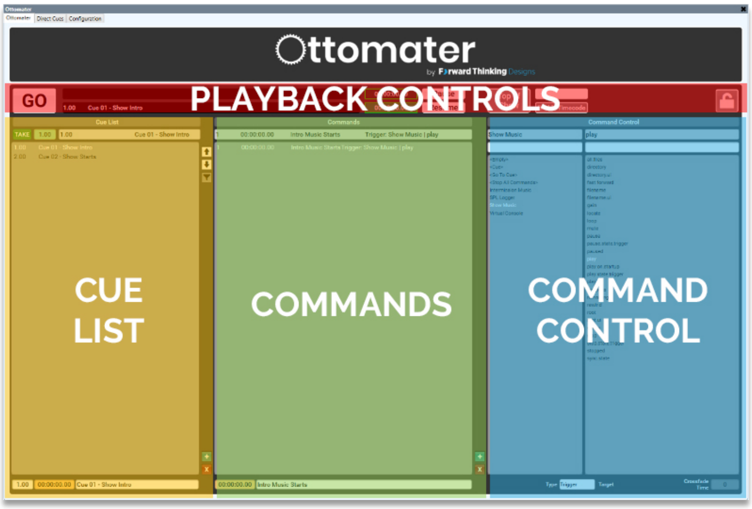
We hope that this walkthrough has been helpful in learning to use Ottomater. Many more technical questions will be addressed in the rest of this manual.

CONTROLS

Ottomater

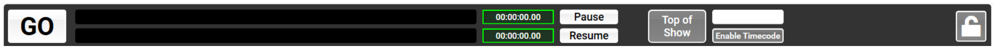


When the plugin is initially opened, no cues or commands will be populated, as shown above.



The UCI is divided into four main sections – Playback Controls, Cue List, Commands, and Command Control.

Playback Controls



GO Button – Plays the currently selected Cue.

Current Cue – Shows the name of the currently playing Cue.

Next Cue – Shows the name of the selected Cue.

Time Elapsed – Shows the time elapsed of the currently playing Cue.

Time Remaining – Shows the time remaining of the selected Cue.

Pause – Pauses Ottomater's internal clock, and consequently the triggering of Cues and Commands.

Resume – Resumes the triggering of Cues and Commands. Manually triggering a Cue will automatically resume playback, as will Timecode triggering a Cue.

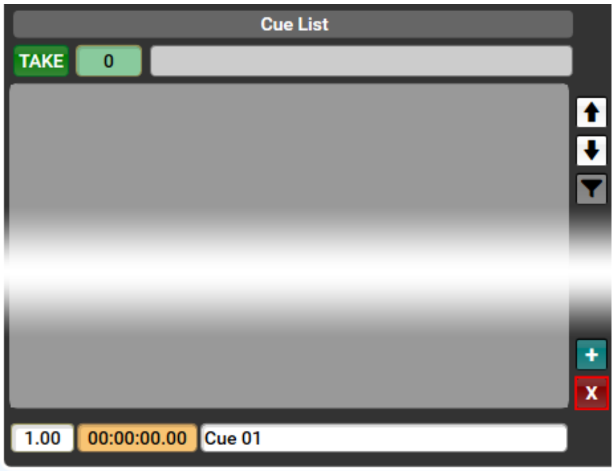
Top of Show – Selects the earliest Cue in the Cue List.

Current Timecode – Shows the Timecode, if enabled.

Enable Timecode – Enables firing Cues by Timecode. When this is enabled, Cues will automatically fire as soon as the Timecode reaches the Cue Timecode.

Lock – The Lock button disables editing of Cues, Commands, and Controls. This does not prevent playing Cues. Use of the Lock button is recommended to prevent accidental changes to the completed show file.

Cue List



Take Cue – Immediately fire the Cue in the Selected Cue field. Functions like the Go button.

Recall Cue – Select a Cue by entering its Cue Number.

Selected Cue – Displays the Cue Name of the currently selected Cue.

Cue List – The list of Cues in the Show. By default, displays up to 36 Cues at a time but this can be limited with the Enable Cue Limit button.

Go To Previous Cue – Selects the Cue before the currently selected Cue.

Go To Next Cue – Selects the Cue after the currently selected Cue.

Enable Cue Limit – Toggles between displaying the full Cue List and a reduced set of Cues. The number of previous and next Cues to be displayed can be set on the Configuration tab.

Insert Cue –Add a blank Cue to the Cue List.

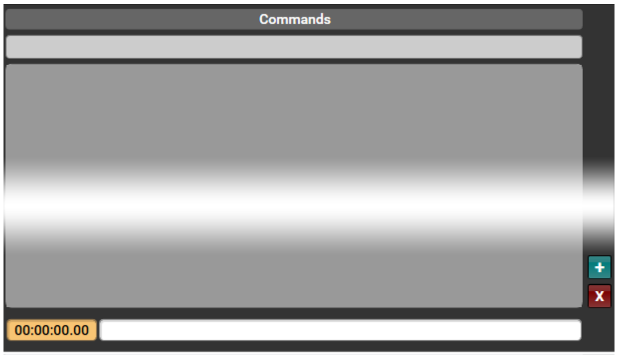
Delete Cue – Delete the selected Cue and any Commands contained within it.

Edit Cue Number – Sets the Cue Number of the selected Cue.

Edit Cue Timecode – Sets the Timecode at when to trigger the selected Cue.

Cue Name Field – Sets the Name of the selected Cue.

Commands



Selected Command – Displays the name of the currently selected Command.

Command List – The list of Commands in the currently selected Cue.

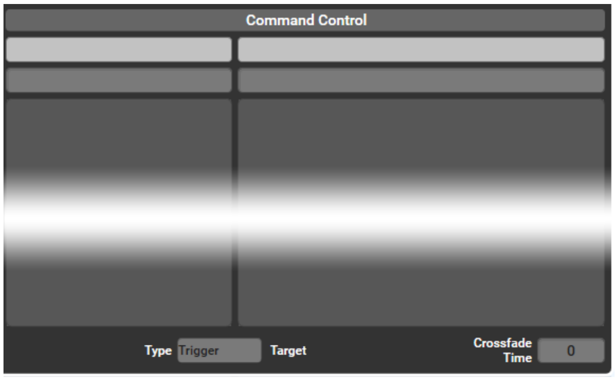
Insert Command – Add a Command to the Commands list.

Delete Command – Delete the selected Command.

Edit Command Time – Sets the time delay, relative to the containing Cue, after which the selected Command will fire automatically. This is relative to the containing Cue and does not correspond directly to Timecode.

Edit Command Name – Sets the Name of the selected Command.>

Command Control List



Selected Component – Displays the name of the currently selected Component.

Search Components – Filters the Component List by the entered text.

Component List – The Components that Ottomater can control. Component List will display only Components which match the text entered in Search Components.

Selected Control – Displays the name of the currently selected Control.

Search Controls – Filter the Control List by the entered text.

Control List –The Controls for the selected Component. Control List will display only Controls which match the text entered in Search Controls.

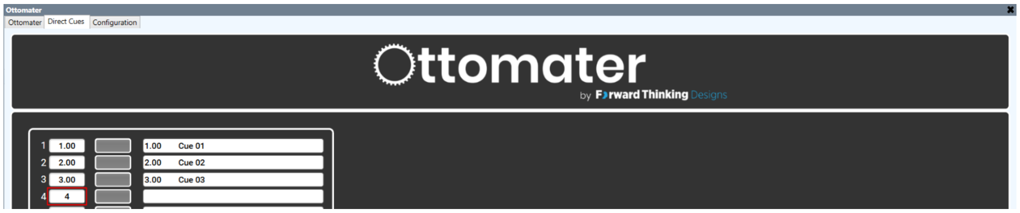
Target Type – Sets the Q-SYS data type for the selected Control. Ottomater uses the Target Type when sending instructions to the Control. If the Target Type does not match the Control, the Command may not function or may behave unexpectedly. Options include:

- Trigger
- Boolean
- Value
- Position
- Text

Target – Sets the desired state of the selected Control at the end of the Command. Available for the Target Types of Boolean, Value, Position, and Text.

Crossfade Time – Sets the duration, in seconds, for the transition from the starting state to the Target of the selected Control to the Target. Available for the Target Types of Value and Position.

Direct Cues



1 1.00 1.00 Cue 01
2 2.00 2.00 Cue 02
3 3.00 3.00 Cue 03
4 4

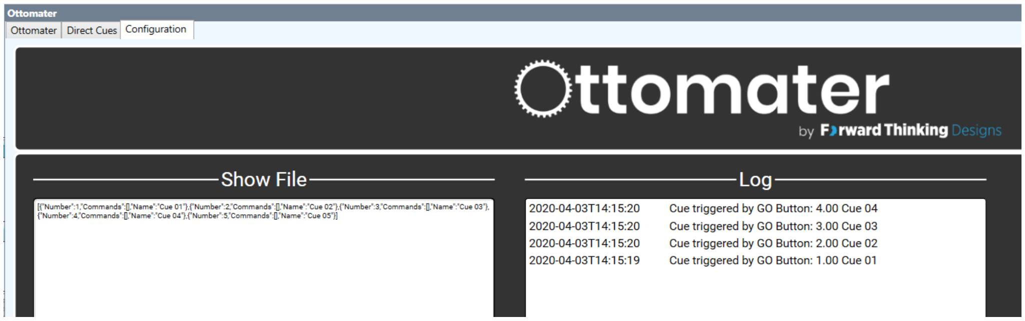
The number of Direct Cues displayed is determined by the value of the Direct Cues Property.

Direct Cue Number – Sets the Cue Number which the respective Direct Cue Take references. This behaves like an address. If this does not match a Cue Number, the Direct Cue will have no effect.

Direct Cue Take – A Trigger Button that fires the Cue referenced by the associated Direct Cue Number. This can be dragged on to a UCI or connected via Control Pin.

Direct Cue Name –Displays the name of the Cue that is referenced by the Direct Cue Number.

Configuration



```
[{"Number":1,"Commands":[],"Name":"Cue 01"},{"Number":2,"Commands":[],"Name":"Cue 02"},{"Number":3,"Commands":[],"Name":"Cue 03"},
{"Number":4,"Commands":[],"Name":"Cue 04"},{"Number":5,"Commands":[],"Name":"Cue 05"}]
```

```
2020-04-03T14:15:20    Cue triggered by GO Button: 4.00 Cue 04
2020-04-03T14:15:20    Cue triggered by GO Button: 3.00 Cue 03
2020-04-03T14:15:20    Cue triggered by GO Button: 2.00 Cue 02
2020-04-03T14:15:19    Cue triggered by GO Button: 1.00 Cue 01
```

The Configuration tab has several outputs, including the Show file.

Show File – Ottomater outputs the complete Show configuration in text to this text field. The output can be copied to the system clipboard and then pasted to a .txt file and saved to disk as storage or backup. The output can be pasted back in to retrieve the Show configuration.

Log – Displays Ottomater's log. If *Log Cues* and/or *Log Commands* is set to Yes, a log line will be written here each time a Cue or Command is triggered.

Cue List Limit – Set the number of Cues to display when *Enable Cue Limit* is enabled.

APPENDIX A: CONTROL PINS

Direct Cue

Pin Name	Control Type	Value Range	Pin Direction
Name	String		Output
Number	Float	0.0 – 9999.0	Input / Output
Take	Trigger		Input / Output

General

Pin Name	Control Type	Value Range	Pin Direction
Cue Limit Bottom	Integer	0 – 100	Input / Output
Cue Limit Top	Integer	0 – 100	Input / Output
Current Cue	String		Output
Current Timecode	String		Output
Enable Cue Limit	Boolean		Input / Output
Enable Timecode	Boolean		Input / Output
External Time[1]	Float	0.0 – 100.0	Input
External Timecode[2]	String	HH:MM:SS.FF	Input
Go	Trigger		Input / Output
Go To Next Cue	Trigger		Input / Output
Go To Previous Cue	Trigger		Input / Output
Lock	Boolean		Input / Output
Next Cue	String		Output
Pause	Trigger		Input / Output
Recall Cue	Float	0.0 – 9999.0	Input / Output
Resume	Trigger		Input / Output
Selected Cue	String		Output
Show File	String		Input / Output
Take Cue	Trigger		Input / Output
Time Elapsed	String		Output
Time Remaining	String		Output
Top of Show	Trigger		Input / Output

[1] When *Timecode* is set to "Seconds."

[2] When *Timecode* is set to "Timecode" or "Time of Day"

SUPPORT

For plugin support and feedback please contact us at:

ForwardThinkingDesigns.com

support@forwardthinkingdesigns.com

+1 407-850-8093